305-CD-006-001

EOSDIS Core System Project

Release A SDPS Interoperability Subsystem Design Specification for the ECS Project

July 1995

Hughes Information Technology Corporation
Landover, MD

Release A SDPS Interoperability Subsystem Design Specification for the ECS Project

July 1995

Prepared Under Contract NAS5-60000
CDRL Item #046

APPROVED BY

Parag N. Ambardekar /s/ 7/28/95

Parag Ambardekar, Release A CCB Chairman Date
EOSDIS Core System Project

Hughes Information Technology Corporation
Landover, Maryland

This page intentionally left blank.

Preface

This document is one of sixteen comprising the detailed design specifications of the SDPS and CSMS subsystem for Release A of the ECS project. A complete list of the design specification documents is given below. Of particular interest are documents number 305-CD-004, which provides an overview of the subsystems and 305-CD-018, the Data Dictionary, for those reviewing the object models in detail. A Release A SDPS and CSMS CDR Review Guide (510-TP-002) is also available.

The SDPS and CSMS subsystem design specification documents for Release A of the ECS Project include:

305-CD-004	Release A Overview of the SDPS and CSMS Segment System Design Specification
305-CD-005	Release A SDPS Client Subsystem Design Specification
305-CD-006	Release A SDPS Interoperability Subsystem Design Specification
305-CD-007	Release A SDPS Data Management Subsystem Design Specification
305-CD-008	Release A SDPS Data Server Subsystem Design Specification
305-CD-009	Release A SDPS Ingest Subsystem Design Specification
305-CD-010	Release A SDPS Planning Subsystem Design Specification
305-CD-011	Release A SDPS Data Processing Subsystem Design Specification
305-CD-012	Release A CSMS Segment Communications Subsystem Design Specification
305-CD-013	Release A CSMS Segment Systems Management Subsystem Design Specification
305-CD-014	Release A GSFC Distributed Active Archive Center Implementation
305-CD-015	Release A LaRC Distributed Active Archive Center Implementation
305-CD-016	Release A MSFC Distributed Active Archive Center Implementation
305-CD-017	Release A EROS Data Center Distributed Active Archive Center Implementation
305-CD-018	Release A Data Dictionary for Subsystem Design Specification
305-CD-019	Release A System Monitoring and Coordination Center Implementation

Object models presented in this document have been exported directly from CASE tools and in some cases contain too much detail to be easily readable within hard copy page constraints. The reader is encouraged to view these drawings on line using the Portable Document Format (PDF) electronic copy available via the ECS Data Handling System (ECS) at URL http://edhs1.gsfc.nasa.gov.

This document is a contract deliverable with an approval code 2. As such, it does not require formal Government approval, however, the Government reserves the right to request changes within 45 days of the initial submittal. Once approved, contractor changes to this document are handled in accordance with Class I and Class II change control requirements described in the EOS Configuration Management Plan, and changes to this document shall be made by document change notice (DCN) or by complete revision.

Any questions should be addressed to:

Data Management Office
The ECS Project Office
Hughes Information Technology Corporation
1616 McCormick Drive
Landover, MD 20785

Abstract

This document presents the design of the Interoperability Subsystem of the Earth Observing System Data and Information System (EOSDIS) Core System (ECS). It defines the Interoperability Subsystem's Release A CSCI interfaces and structures, as well as subsystem design based on Level 4 requirements.

Keywords: SDPS, Interoperability, CSCI, HWCI, Advertising, Advertisement, Gateway, WAIS, HTTP, HTML, DBMS

This page intentionally left blank.

Change Information Page

List of Effective Pages

Page Number	Issue
Title	Final
iii through viii	Final
1-1 through	Final
2-1 through 2-4	Final
3-1 through 3-4	Final
4-1 through 4-36	Final
AB-1 through AB-6	Final

Document History

Document Number	Status/Issue	Publication Date	CCR Number
305-CD-006-001	Final	July 1995	95-0468

This page intentionally left blank.

Contents

Preface

Abstract

1. Introduction

2. Related Documents

3. Interoperability Subsystem Overview

4. ADSRV - Advertising Service CSCI

Figures

Tables

Abbreviations and Acronyms

1. Introduction

1.1 Identification

This Release A SDPS Interoperability Subsystem Design Specification for the ECS Project, Contract Data Requirement List (CDRL) Item 046, with requirements specified in Data Item Description (DID) 305/DV2, is a required deliverable under the Earth Observing System Data and Information System (EOSDIS) Core System (ECS), Contract NAS5-60000. This publication is part of a series of documents comprising the Science and Communications Development Office design specification for the Communications and System Management Segment (CSMS) and the Science and Data Processing Subsystem (SDPS) for Release A.

1.2 Scope

The Release A SDPS Interoperability Subsystem Design Specification defines the progress of the Interoperability subsystem design. It defines the Interoperability Subsystem computer software and hardware architectural design, as well as subsystem design based on Level 4 requirements.

This subsystem is on an incremental development track. It is released in and reviewed in the form of Evaluation Packages (EP), and is therefore, not part of the formal Release A Critical Design Review. The overview material for these components has been included in this document for information purposes only.

This document reflects the June 21, 1995 Technical Baseline maintained by the Contracts Configuration Control Board in accordance with ECS Technical Direction No. 11 dated December 6, 1994.

1.3 Document Organization

The document is organized to describe the Release A SDPS Interoperability Subsystem design as follows:

- Section 1 provides information regarding the identification, scope, organization and status.
- Section 2 provides a listing of related documents.
- Section 3 provides an overview of the Subsystem, focusing on the high-level design concepts. This provides general background information to put interoperability into context.
- Section 4 contains the structure of the Computer Software Configuration Items (CSCI) comprising the Interoperability Subsystem.
- The section Abbreviations and Acronyms contains an alphabetized list of the definitions for abbreviations and acronyms used in this volume.

1.4 Status and Schedule

This submittal of DID 305/DV3 meets the milestone specified in the Contract Data Requirements List (CDRL) of NASA Contract NAS5-60000. The submittal was reviewed during the SDPS Preliminary Design Review (PDR) and reflects changes to the design which resulted from that review. The PDR also triggered a number of follow-up actions in response to Review Item Descrepencies (RID) the results of which will be incorporated into the Critical Design Review (CDR) version of this document.

This page intentionally left blank.

2. Related Documents

2.1 Parent Documents

The parent document is the document from which the scope and content of this Interoperability Subsystem Design Specification is derived.

194-207-SE1-001	System Design Specification for the ECS Project

2.2 Applicable Documents

The following documents are referenced within this SDPS Design Specification, or are directly applicable, or contain policies or other directive matters that are binding upon the content of this volume.

209-CD-001-001	Interface Control Document Between EOSDIS Core System (ECS) and the NASA Science Internet
209-CD-002-001	Interface Control Document Between EOSDIS Core System (ECS) and ASTER Ground Data System
209-CD-003-001	Interface Control Document Between EOSDIS Core System (ECS) and EOS-AM Project for AM-1 Spacecraft Analysis Software
209-CD-004-001	Data Format Control Document for the Earth Observing System (EOS) AM-1 Project Data Base
209-CD-005-002	Interface Control Document Between EOSDIS Core System (ECS) and Science Computing Facilities (SCF)
209-CD-006-002	Interface Control Document Between EOSDIS Core System (ECS) and National Oceanic and Atmospheric Administration (NOAA) Affiliated Data Center (ADC)
209-CD-007-002	Interface Control Document Between EOSDIS Core System (ECS) and TRMM Science Data and Information System (TSDIS)
209-CD-008-002	Interface Control Document Between EOSDIS Core System (ECS) and the Goddard Space Flight Center (GSFC) Distributed Active Archive Center (DAAC)
209-CD-009-002	Interface Control Document Between EOSDIS Core System (ECS) and the Marshall Space Flight Center (MSFC) Distributed Active Archive Center (DAAC)
209-CD-011-002	Interface Control Document Between EOSDIS Core System (ECS) and the Version 0 System
305-CD-003-002	Communications and System Management Segment (CSMS) Design Specification for the ECS Project
308-CD-001-004	Software Development Plan for the ECS Project

313-CD-004-001	Release A CSMS/SDPS Internal Interface Control Document for the ECS Project
423-41-03	Goddard Space Flight Center, EOSDIS Core System (ECS) Contract Data Requirements Document

2.3 Information Documents Not Referenced

The following documents, although not referenced herein and/or not directly applicable, do amplify and clarify the information presented in this document. These documents are not binding on the content of the SDPS Design Specifications.

205-CD-002-002	Science User's Guide and Operations Procedure Handbook for the ECS Project. Part 4: Software Developer's Guide to Preparation, Delivery, Integration, and Test with ECS
206-CD-001-002	Version 0 Analysis Report for the ECS Project
209-CD-010-001	Interface Control Document Between EOSDIS Core System (ECS) and the Langley Research Center (LaRC) Distributed Active Archive Center (DAAC) Draft
194-302-DV2-001	ECS Facilities Plan for the ECS Project
101-303-DV1-001	Individual Facility Requirements for the ECS Project, Preliminary
194-317-DV1-001	Prototyping and Studies Plan for the ECS Project
318-CD-003-XXX	Prototyping and Studies Progress Report for the ECS Project (monthly)
333-CD-003-001	SDP Toolkit Users Guide for the ECS Project
601-CD-001-002	Maintenance and Operations Management Plan for the ECS Project
604-CD-001-004	Operations Concept for the ECS Project: Part 1 -- ECS Overview
604-CD-002-001	Operations Concept for the ECS project: Part 2B -- ECS Release B, Annotated Outline
604-CD-003-001	ECS Operations Concept for the ECS Project: Part 2A -- ECS Release A, Final
101-620-OP2-001	List of Recommended Maintenance Equipment for the ECS Project
194-703-PP1-001	System Design Review (SDR) Presentation Package for the ECS Project
193-801-SD4-001	PGS Toolkit Requirements Specification for the ECS Project
194-813-SI4-002	Planning and Scheduling Prototype Results Report for the ECS Project
194-813-SI4-003	DADS Prototype One FSMS Product Operational Evaluation
194-813-SI4-004	DADS Prototype One STK Wolfcreek 9360 Automated Cartridge System Hardware Characterization Report
813-RD-009-001	DADS Prototype Two Multi-FSMS Product Integration Evaluation
828-RD-001-002	Government Furnished Property for the ECS Project
193-WP-118-001	Algorithm Integration and Test Issues for the ECS Project

193-WP-611-001	Science-based System Architecture Drivers for the ECS Project, Revision 1.0
193-WP-623-001	ECS Evolutionary Development White Paper
194-WP-901-002	EOSDIS Core System Science Information Architecture, White Paper, Working Paper
194-WP-902-002	ECS Science Requirements Summary, White Paper, Working Paper
194-WP-904-002	Multi-Track Development for the ECS Project, White Paper, Working Paper
194-WP-913-003	User Environment Definition for the ECS Project, White Paper, Working Paper
194-WP-914-001	CORBA Object Request Broker Survey for the ECS Project, White Paper, Working Paper
194-WP-918-001	DADS Prototype One FSMS Product Operational Evaluation, White Paper, Draft Report
194-WP-925-001	Science Software Integration and Test, White Paper, Working Paper
420-WP-001-001	Maximizing the Use of COTS Software in the SDPS SDS Software Design, White Paper
193-TP-626-001	GCDIS/UserDIS Study ECS Technical Paper, Draft 0.2
194-TP-266-002	Data Distribution Architecture Logical Object Model (LOM) for the ECS Project, Version 2.01
194-TP-267-001	Data Server Architecture Logical Object Model (LOM) for the ECS Project, Version 2.00
194-TP-313-001	ECS User Characterization Methodology and Results
194-TP-316-002	Data Compression Study for the ECS Project
194-TP-548-001	User Scenario Functional Analysis [for the ECS Project]
194-TP-569-001	PDPS Prototyping at ECS Science and Technology Laboratory, Progress Report #4
222-TP-003-006	Release Plan Content Description for the ECS Project
430-TP-001-001	SDP Toolkit Implementation with Pathfinder SSM/I Precipitation Rate Algorithm, Technical Paper
440-TP-001-001	Science Data Server Architecture Study [for the ECS Project]
none	Hughes Training, Inc., ECS User Interface Style Guide, White Paper, Version 4.0
423-16-01	Goddard Space Flight Center, Data Production Software and Science Computing Facility (SCF) Standards and Guidelines
423-41-02	Goddard Space Flight Center, Functional and Performance Requirements Specification for the Earth Observing System Data and Information System (EOSDIS) Core System

540-022	Goddard Space Flight Center, Earth Observing System (EOS) Communications (Ecom) System Design Specification
560-EDOS-0211.0001	Goddard Space Flight Center, Interface Requirements Document Between EDOS and the EOS Ground System (EGS)

3. Interoperability Subsystem Overview

3.1 Subsystem Overview

The SDPS Interoperability Subsystem includes one CSCI, the Advertising Service. Users access this subsystem to search for and locate both ECS and non-ECS services, providers, and data. The subsystem receives advertising information from ECS and non-ECS service providers, and provides interfaces for users to access and subscribe to the advertising information.

3.2 Subsystem Structure

The Interoperability Subsystem for Release A is composed of one CSCI, the Advertising Service (ADSRV), a software component. It provides advertisements about both ECS and non-ECS services, providers and data.. The HWCI is shared with Data Management subsystem and the details can be found in Section 5 of Data Management Subsytem Design Sepcification.

The Interoperability Subsystem context diagram is presented in Figure 3.2-1 It illustrates the relationships between the Interoperability subsystem and the other SDPS subsystems.

- The subsystem accepts advertisements, subscriptions, and search requests from the Client subsystem. The Client subsystem provides the interface to users.
- The subsystem accepts search requests from the Ingest subsystems.
- The subsystem imports directory level information from the Global Change Master Directory (GCMD) via the GCMD export files.
- The subsystem receives ECS valid mapping results from Gateway subsystem.
- The subsystem also accepts advertisements and subscriptions from the Data Server subsystem.
- Advertisements and subscriptions are also received from non-ECS service providers such as International Partners, Science Computing Facilities, Affiliated Data Centers, etc.

The Interoperability Subsystem interfaces are listed in Table 3.2-1.

3.3 Subsystem Design Rationale

This Interoperability Subsystem is proceeding along the incremental development track. The general design emphasizes a user friendly interface for a scalable, evolvable, and easily maintained subsystem. The subsystem design is driven primarily by the following drivers:

- Provide an integrated view of the data and service network.
- Evolve Interoperability capabilities as technology and research progress.
- Allow for an extensible provider network.
- Provide a framework for the "publish and subscribe paradigm" in order to access data and services.

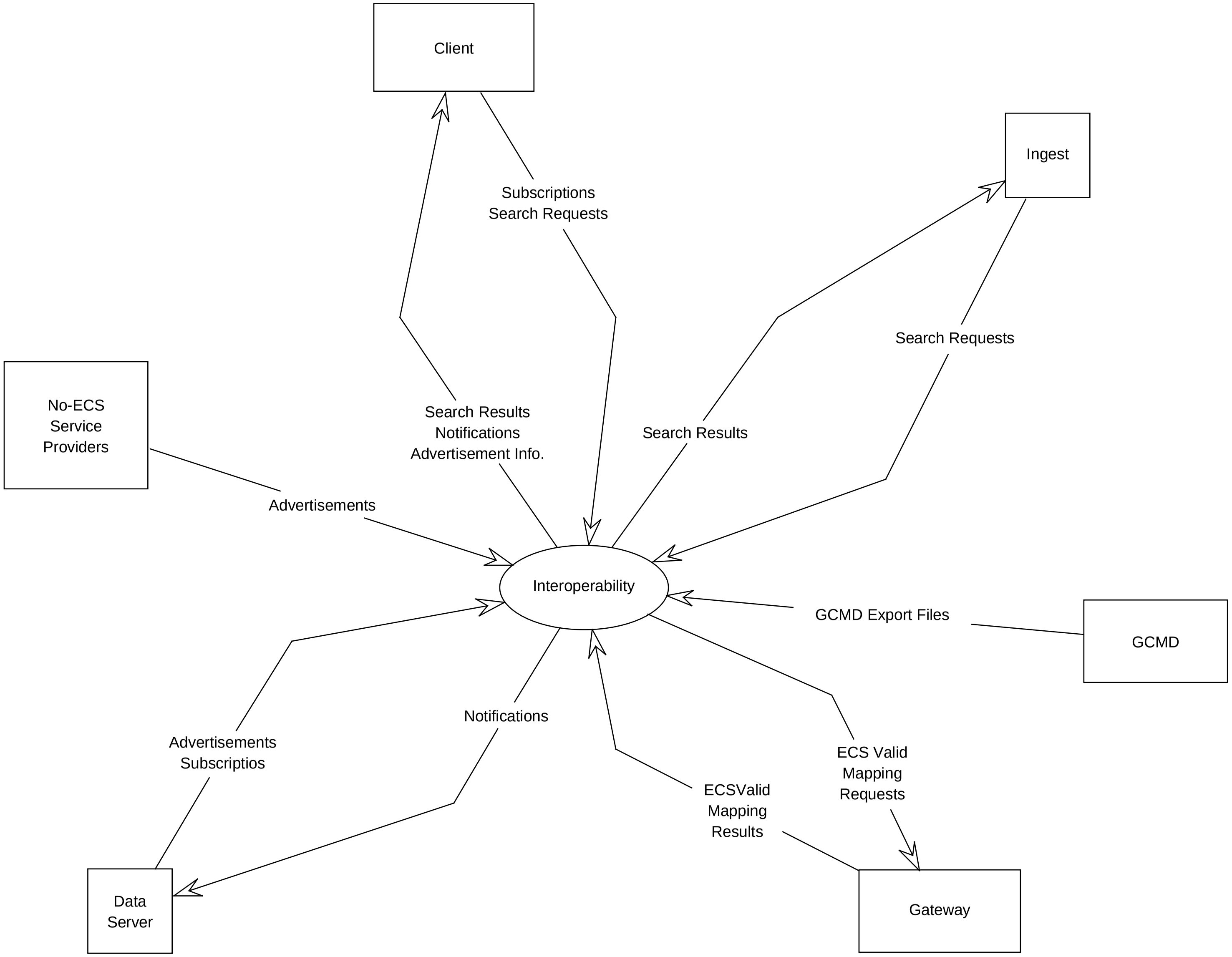


Figure 3.2-1. Interoperability Subsystem Context

Table 3.2-1. Interoperability Subsystem Interfaces

Source	Destination	Data Types	Data Volume	Frequency
Client	Interoperability	Advertisements	low	as requested
Client	Interoperability	Subscriptions	low	as requested
Client	Interoperability	Search Requests	low	as requested
Ingest	Interoperability	Search Requests	low	as required
GCMD	Interoperability	GCMD Export Files	low	as required
Gateway	Interoperability	ECS Valid Mapping Results	low	in response to requests
Data Server	Interoperability	Advertisements	low	as required
Data Server	Interoperability	Subscriptions	low	as required
Non-ECS Service Providers	Interoperability	Advertisements	low	as required
Non-ECS Service Providers	Interoperability	Subscriptions	low	as required
Interoperability	Client	Search Results	low	in response to requests
Interoperability	Client	Advertisement info	low	in response to requests
Interoperability	Client	Notifications	low	in response to subscriptions
Interoperability	Ingest	Search Results	low	in response to requests
Interoperability	Gateway	ECS Valid Mapping Requests	low	as requested
Interoperability	Data Server	Notifications	low	in response to subscriptions
Interoperability	Non-ECS Service Providers	Notifications	low	in response to subscriptions

In the table, where an exact number is unavailable, the data volume is estimated as low (less than 1 MB), medium (between 1 MB and 1 GB), or high (greater than 1 GB) per use defined in the frequency column. The frequency information will be updated as the interfaces are fully defined.

This page intentionally left blank.

4. ADSRV - Advertising Service CSCI

4.1 CSCI Overview

The Advertising Service provides the interfaces needed to support Client defined interactive browsing and searching of advertisements. Although there will be a single format for submitting advertisements to the service, advertisements should be accessible via several different interfaces to support database searching, text searching, and hyper linked access and retrieval according to several different viewing styles (e.g., plain ASCII text, interactive form, or HTML document).

A data server or other provider will advertise its data collections and services with the Advertising Service. The advertisement will include a listing of all products (and other Earth Science Data Types) available in the collection and a set of product attributes. Advertisements include directory level metadata, therefore, the attributes reflected in the advertising service include the ECS Core Metadata Directory-Level attributes. The workbench will send user queries which access only directory level metadata directly to the advertising service (rather than sending it as a distributed query to the various sites which provided the advertising information). A user who wishes to find out what data sets are available on the network can search (i.e., formulate a query) or browse (i.e., navigate through hyperlinked pages of advertisements) the advertising information. Both types of 'directory searching' are available on the user's desktop; the user can choose whichever approach is most convenient in the current work context.

Since the ADSRV CSCI is on the incremental track of development, requirements, schedule, scenarios, issues and design are documented in a Software Development File (SDF) for ADSRV.

4.2 CSCI Context

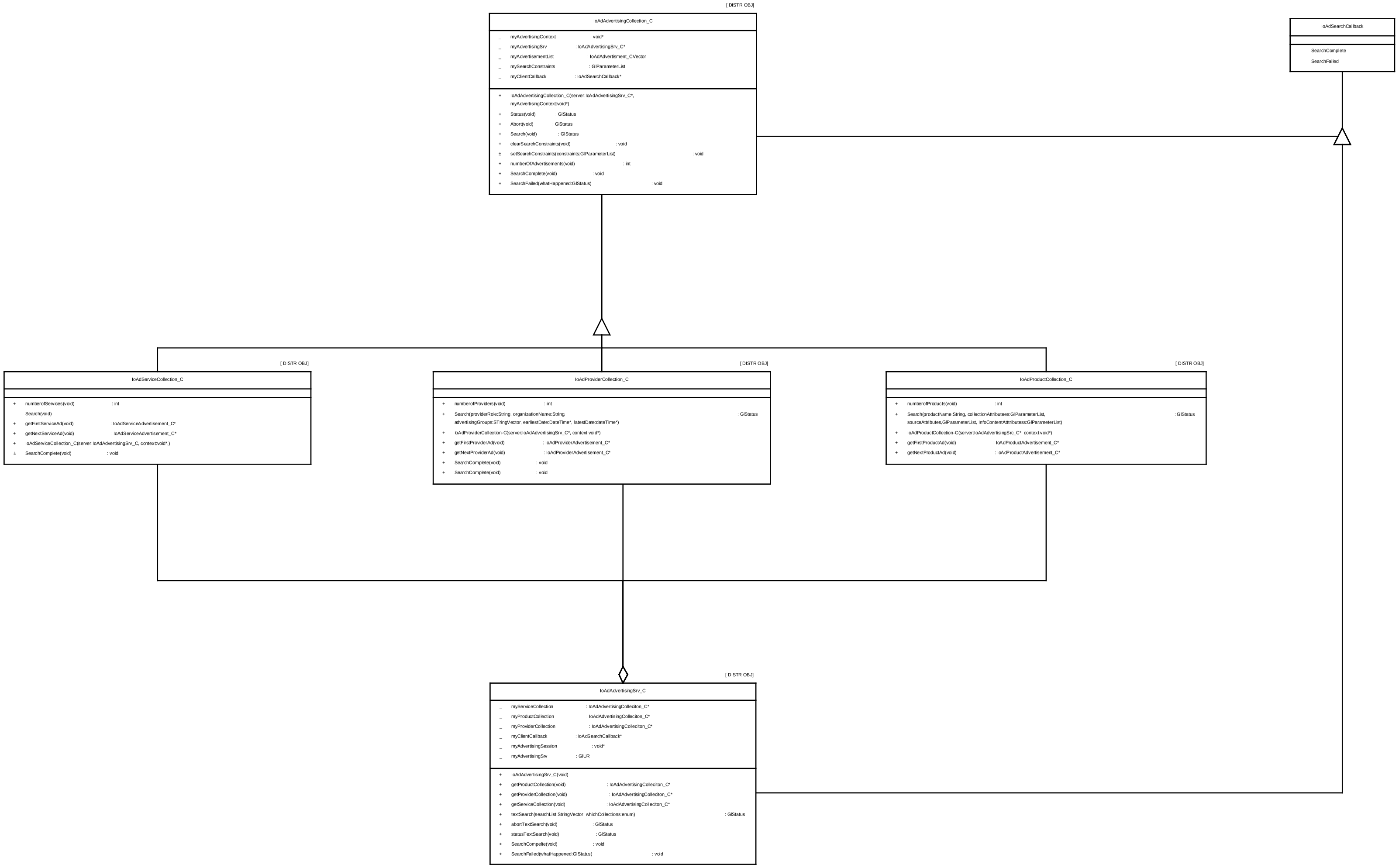
ADSRV CSCI is the only CSCI in the Interoperability Subsystem. Therefore the context of the CSCI is identical to the subsystem context which is shown in Fig 3.2-1.

4.3 CSCI Object Model

This section presents the object model for the interface between the ADSRV CSCI and the formal track CSCIs that were shown in the subsystem interfaces of section 3. The object models have been separated into an advertisement search interface (section 4.3.1) and an advertisement submission interface (section 4.3.2).

4.3.1 Search Interface Object Model

The Search Interface model provides interface objects to be used in searching for advertisements. The interface is composed of a advertising server interface (IoAdAdvertisingSrv_C) and collection interfaces for each of the three types of advertisements. The three collection interfaces are all specialized derived types from a generic advertisement collection base class (IoAdAdvertisingCollection_C). Each of the derived collection types provides specialized operations (e.g. Search) of the base class virtual operations. The inherited collections structure is identical to the advertisement inheritance structure identified in the Submission Interface model of section 4.3.2.


[DISTR OBJ]
IoAdAdvertisingCollection_C
_ myAdvertisingContext : void*
_ myAdvertisingSrv : IoAdAdvertisingSrv_C*
_ myAdvertisementList : IoAdAdvertisment_CVector
_ mySearchConstraints : GlParameterList
_ myClientCallback : IoAdSearchCallback*
+ IoAdAdvertisingCollection_C(server:IoAdAdvertisingSrv_C*, myAdvertisingContext:void*)
+ Status(void) : GlStatus
+ Abort(void) : GlStatus
+ Search(void) : GlStatus
+ clearSearchConstraints(void) : void
± setSearchConstraints(constraints:GlParameterList) : void
+ numberOfAdvertisements(void) : int
+ SearchComplete(void) : void
+ SearchFailed(whatHappened:GlStatus) : void
IoAdSearchCallback
SearchComplete
SearchFailed
[DISTR OBJ]
IoAdServiceCollection_C
+ numberofServices(void) : int
Search(void)
+ getFirstServiceAd(void) : IoAdServiceAdvertisement_C*
+ getNextServiceAd(void) : IoAdServiceAdvertisement_C*
+ IoAdServiceCollection_C(server:IoAdAdvertisingSrv_C, context:void*,)
± SearchComplete(void) : void
[DISTR OBJ]
IoAdProviderCollection_C
+ numberofProviders(void) : int
+ Search(providerRole:String, organizationName:String, advertisingGroups:STringVector, earliestDate:DateTime*, latestDate:dateTime*) : GlStatus
+ IoAdProviderCollection-C(server:IoAdAdvertisingSrv_C*, context:void*)
+ getFirstProviderAd(void) : IoAdProviderAdvertisement_C*
+ getNextProviderAd(void) : IoAdProviderAdvertisement_C*
+ SearchComplete(void) : void
+ SearchComplete(void) : void
[DISTR OBJ]
IoAdProductCollection_C
+ numberofProducts(void) : int
+ Search(productName:String, collectionAttributees:GlParameterList, sourceAttributes,GlParameterList, InfoContentAtttributess:GlParameterList) : GlStatus
+ IoAdProductCollection-C(server:IoAdAdvertisingSrc_C*, context:void*)
+ getFirstProductAd(void) : IoAdProductAdvertisement_C*
+ getNextProductAd(void) : IoAdProductAdvertisement_C*
[DISTR OBJ]
IoAdAdvertisingSrv_C
_ myServiceCollection : IoAdAdvertisingColleciton_C*
_ myProductCollection : IoAdAdvertisingColleciton_C*
_ myProviderCollection : IoAdAdvertisingColleciton_C*
_ myClientCallback : IoAdSearchCallback*
_ myAdvertisingSession : void*
_ myAdvertisingSrv : GlUR
+ IoAdAdvertisingSrv_C(void)
+ getProductCollection(void) : IoAdAdvertisingColleciton_C*
+ getProviderCollection(void) : IoAdAdvertisingColleciton_C*
+ getServiceCollection(void) : IoAdAdvertisingColleciton_C*
+ textSearch(searchList:StringVector, whichCollections:enum) : GlStatus
+ abortTextSearch(void) : GlStatus
+ statusTextSearch(void) : GlStatus
+ SearchCompelte(void) : void
+ SearchFailed(whatHappened:GlStatus) : void

4.3.1-1 IoAd_SearchIF Object Model Diagram

4.3.1.1 IoAdAdvertisingCollection_C Class

Parent Class: IoAdSearchCallback
Public: YesDistributed Object: Yes
Purpose and Description:
This class is a generic base class for the interface to collections of advertisements. This class impliments the common functionality of all advertisement collections. Specialized versions of the collection class are provided for Service, Product, and Provider advertisement collections.

Attributes:

myAdvertisementList - stores the list of advertisements that are in the current collection.
Data Type: IoAdAdvertisment_CVector
Privilege: Private
Default Value:

myAdvertisingContext - stores the session context for managing communications with the advertising DBMS application server.
Data Type: void*
Privilege: Private
Default Value:

myAdvertisingSrv - stores a pointer to the IoAdAdvertisingSrv_C interface object.
Data Type: IoAdAdvertisingSrv_C*
Privilege: Private
Default Value:

myClientCallback - stores a pointer to the client callback object (IoAdSearchCallback) to be invoked when an advertising search completes.
Data Type: IoAdSearchCallback*
Privilege: Private
Default Value:

mySearchConstraints -
Data Type: GlParameterList
Privilege: Private
Default Value:

Operations:

Abort - this operation aborts any active search operation that is being performed on the advertising collection.
Arguments: void
Return Type: GlStatus
Privilege: Public

IoAdAdvertisingCollection_C - this operation initializes the IoAdAdvertisingCollection_c object.
Arguments: server:IoAdAdvertisingSrv_C*, myAdvertisingContext:void*
Return Type: Void
Privilege: Public

Search - this operation initiates an attribute search operation on the current advertising collection. The actual search operation is performed by the advertising DBMS application server.
Arguments: void
Return Type: GlStatus
Privilege: Public

SearchComplete - this virtual operation is invoked whenever an attribute search operation completes successfully.
Arguments: void
Return Type: void
Privilege: Public

SearchFailed - this virtual operation is invoked whenever an attribute search operation fails.
Arguments: whatHappened:GlStatus
Return Type: void
Privilege: Public

Status - this operation returns the status of an attribute search operation that is in progress.
Arguments: void
Return Type: GlStatus
Privilege: Public

clearSearchConstraints - this operation clears the current set of search constraints.
Arguments: void
Return Type: void
Privilege: Public

numberOfAdvertisements - this operation returns the number of advertisements in the

current collection.
Arguments: void
Return Type: int
Privilege: Public

setSearchConstraints - this operation is used to set a new set of attribute search constraints.
Arguments: constraints:GlParameterList
Return Type: void
Privilege: Protected

Associations:

The IoAdAdvertisingCollection_C class has associations with the following classes:
None

4.3.1.2 IoAdAdvertisingSrv_C Class

Parent Class: IoAdSearchCallback
Public: YesDistributed Object: Yes
Purpose and Description:
IoAdAdvertisingSrv_C is an aggregate class of service, provider, and product advertisement. It manages the collections of service, provider, and product advertisement.

Attributes:

myAdvertisingSession - stores the advertising session context pointer that is returned from the advertising DBMS application server.
Data Type: void*
Privilege: Private
Default Value:

myAdvertisingSrv - stores the universal reference of the advertising DBMS application server.
Data Type: GlUR
Privilege: Private
Default Value:

myClientCallback - stores a pointer to the IoAdSearchCallback object to be invoked when an a text search operation completes.

Data Type: IoAdSearchCallback*
Privilege: Private
Default Value:

myProductCollection - stores a pointer to the product advertisement collection.
Data Type: IoAdAdvertisingColleciton_C*
Privilege: Private
Default Value:

myProviderCollection - stores a pointer to the provider advertisement collection.
Data Type: IoAdAdvertisingColleciton_C*
Privilege: Private
Default Value:

myServiceCollection - stores a pointer to the service advertisement collection.
Data Type: IoAdAdvertisingColleciton_C*
Privilege: Private
Default Value:

Operations:

IoAdAdvertisingSrv_C - this constructor initializes the IoAdAdvertisingSrv_C object.
Arguments: void
Return Type: Void
Privilege: Public

SearchCompelte - this operation is invoked whenever a text search operation completes successfully. Upon completion the advertising collections are updated, and the client callback is invoked.
Arguments: void
Return Type: void
Privilege: Public

SearchFailed - this operation is invoked when a text search operation fails.
Arguments: whatHappened:GlStatus
Return Type: void
Privilege: Public

abortTextSearch - this operation aborts a text search operation that is in progress.
Arguments: void
Return Type: GlStatus
Privilege: Public

getProductCollection - this operation returns a pointer to the collection of product advertisements.
Arguments: void
Return Type: IoAdAdvertisingColleciton_C*
Privilege: Public

getProviderCollection - this operation returns a pointer to the collection of provider advertisements.
Arguments: void
Return Type: IoAdAdvertisingColleciton_C*
Privilege: Public

getServiceCollection - this operation returns a pointer to the collection of service advertisements.
Arguments: void
Return Type: IoAdAdvertisingColleciton_C*
Privilege: Public

statusTextSearch - this operation returns the status of a text search operation that is in progress.
Arguments: void
Return Type: GlStatus
Privilege: Public

textSearch - this operation intiates a text search operation on the current advertisement collection. The actual search operation is performed by the advertising DBMS application server.
Arguments: searchList:StringVector, whichCollections:enum
Return Type: GlStatus
Privilege: Public

Associations:

The IoAdAdvertisingSrv_C class has associations with the following classes:
None

4.3.1.3 IoAdProductCollection_C Class

Parent Class: IoAdAdvertisingCollection_C
Public: YesDistributed Object: Yes
Purpose and Description:

The IoAdProductCollection_C class is a specialization of the IoAdAdvertisingCollection_C base class. It is responsible for managing access collections of product advertisements (i.e., IoAdProductAdvertisement_C).

Attributes:

All Attributes inherited from parent class

Operations:

IoAdProductCollection-C - This constructor is responsible for initializing the IoADProductCollection_C class.
Arguments: server:IoAdAdvertisingSrc_C*, context:void*
Return Type: Void
Privilege: Public

Search - this operation is a specialization of the virtual operation of the IoAdAdvertisingCollection_C base class. The operation is used to initiate an attribute search of the product collection. The actual search operation is performed by the advertising DBMS application server.
Arguments: productName:String, collectionAttributees:GlParameterList, sourceAttributes,GlParameterList, InfoContentAtttributess:GlParameterList
Return Type: GlStatus
Privilege: Public

getFirstProductAd - this operation returns a pointer to the first IoAdProductAdvertisement_C object in the current collection.
Arguments: void
Return Type: IoAdProductAdvertisement_C*
Privilege: Public

getNextProductAd - this operation returns a pointer to the next IoAdProductAdvertisement_C object in the current collection. Subsequent calls to this operation will iterate through the product advertisements.
Arguments: void
Return Type: IoAdProductAdvertisement_C*
Privilege: Public

numberofProducts - this operation returns the number of product advertisements in the current collection.
Arguments: void
Return Type: int
Privilege: Public

Associations:

The IoAdProductCollection_C class has associations with the following classes:
IoAdAdvertisingSrv_C (Aggregation)

4.3.1.4IoAdProviderCollection_C Class

Parent Class: IoAdAdvertisingCollection_C
Public: YesDistributed Object: Yes
Purpose and Description:
The IoAdProviderCollection_C is a specialization of the IoAdAdvertisingCollection_C base class. It is responsible for managing access to a collection of provider advertisements (i.e., IoAdProviderAdvertisement_C).

Attributes:

All Attributes inherited from parent class

Operations:

IoAdProviderCollection-C - this constructor initializes the IoAdProviderCollection_C class.
Arguments: server:IoAdAdvertisingSrv_C*, context:void*
Return Type: Void
Privilege: Public

Search - this operation is a specialization of the virtual operation of the base class. It is used to initiate an attribute search of the current provider collection. The search operation is performed by the advertising DBMS application server.
Arguments: providerRole:String, organizationName:String, advertisingGroups:STringVector, earliestDate:DateTime*, latestDate:dateTime*
Return Type: GlStatus
Privilege: Public

SearchComplete - this operation is a specialization of the virtual function in the IoAdAdvertisingCollection_C base class. The operation is invoked whenever an attribute search operation has completed.
Arguments: void
Return Type: void
Privilege: Public

getFirstProviderAd - this operation returns a pointer to the first provider advertisement in

the collection.
Arguments: void
Return Type: IoAdProviderAdvertisement_C*
Privilege: Public

getNextProviderAd - this operation returns a pointer to the next IoAdProviderAdvertisement_C object in the collection. Each subsequent call to this operation will iterate through the list of advertisements in the collection.
Arguments: void
Return Type: IoAdProviderAdvertisement_C*
Privilege: Public

numberofProviders - this operation returns the number of provider advertisements in the current collection.
Arguments: void
Return Type: int
Privilege: Public

Associations:

The IoAdProviderCollection_C class has associations with the following classes:
IoAdAdvertisingSrv_C (Aggregation)

4.3.1.5IoAdSearchCallback Class

Parent Class: Not Applicable
Public: YesDistributed Object: No
Purpose and Description:
The IoAdSearchCallback class provides a general framework for notifying client classes when an advertising search has completed. Whenever a search completes, the appropriate operation of the callback class is invoked.

Attributes:

None

Operations:

SearchComplete - This operation is invoked whenever a search is complete.
Arguments:

SearchFailed - This operation is invoked whenever a search fails.
Arguments:

Associations:

The IoAdSearchCallback class has associations with the following classes:
None

4.3.1.6IoAdServiceCollection_C Class

Parent Class: IoAdAdvertisingCollection_C
Public: YesDistributed Object: Yes
Purpose and Description:
The IoAdServiceCollection_C class is a specialization of the IoAdAdvertisingCollection_C class. It is responsible for providing collection managment for service advertisements (IoAdServiceAdvertisement_C).

Attributes:

All Attributes inherited from parent class

Operations:

IoAdServiceCollection_C - this constructor initializes the service advertisement collection.
Arguments: server:IoAdAdvertisingSrv_C, context:void*,
Return Type: Void
Privilege: Public

Search
Arguments: void

SearchComplete - this operation is a specialization of the virtual operation of the IoAdAdvertisingCollection_C class. It is invoked whenever an attribute search of the service collection completes.
Arguments: void
Return Type: void
Privilege: Protected

getFirstServiceAd - this operation returns a pointer to the first IoAdServiceAdvertisement_C in the collection.

Arguments: void
Return Type: IoAdServiceAdvertisement_C*
Privilege: Public

getNextServiceAd - this operation returns a pointer to the next IoAdServiceAdfvertisement_C object in the collection. Each subsequent call to this operation will iterate throught the list of advertisements.
Arguments: void
Return Type: IoAdServiceAdvertisement_C*
Privilege: Public

numberofServices - this operation returns the number of service advertisements in the current collection.
Arguments: void
Return Type: int
Privilege: Public

Associations:

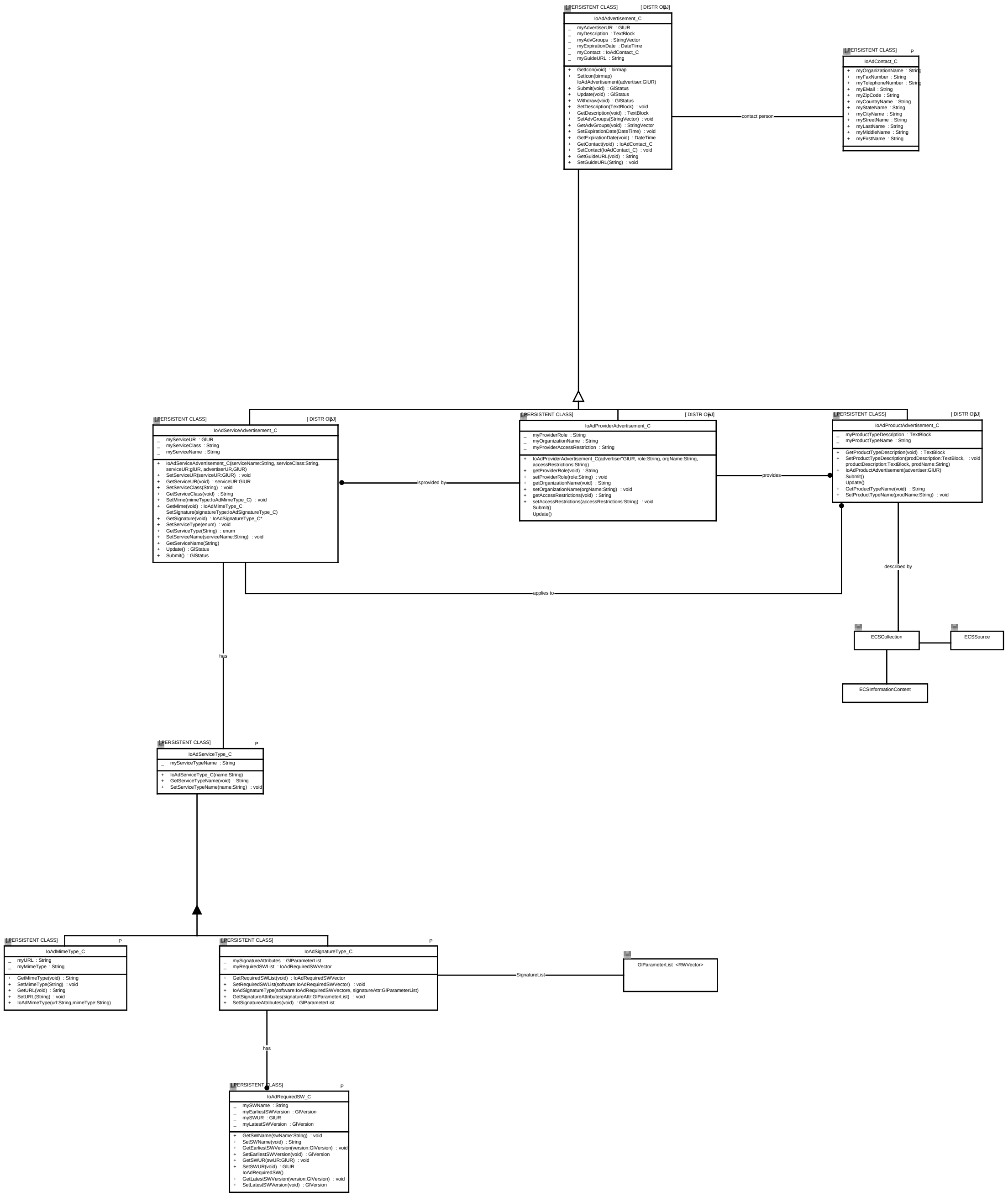
The IoAdServiceCollection_C class has associations with the following classes:
IoAdAdvertisingSrv_C (Aggregation)

4.3.2Submission Interface Object Model

The Submission Interface model provides objects representing service advertisements (i.e. class IoAdServiceAdvertisement_C), provider advertisements (i.e. class IoAdProviderAdvertisement_C) and product advertisements (i.e. class IoAdProviderAdvertisement_C). Each of these advertisement classes is derived from the generalized class (i.e., class IoAdAdvertisement_C). IoAdAdvertisement_C provides the common functionality of all advertisements, including networked communications with the advertising server. In order to submit an advertisement using this interface model, the user would construct an advertisement object of the appropriate type, and submit the advertisement via the Submit operation. Similarly advertisements can be updated, or withdrawn by instantiating an object or the correct type and applying the appropriate operation.

4.3.2.1ECSCollection Class

Parent Class: Not Applicable
Public: NoDistributed Object: No
Purpose and Description:
The ECSCollection class is a placehold that represents the equivalent object from the ECS Core Metadata model. All of the collection specific attributes from the core will be contained in this object.


[PERSISTENT CLASS]
[DISTR OBJ]
P
IoAdAdvertisement_C
_ myAdvertiserUR : GlUR
_ myDescription : TextBlock
_ myAdvGroups : StringVector
_ myExpirationDate : DateTime
_ myContact : IoAdContact_C
_ myGuideURL : String
+ GetIcon(void) : birmap
+ SetIcon(birmap)
IoAdAdvertisement(advertiser:GlUR)
+ Submit(void) : GlStatus
+ Update(void) : GlStatus
+ Withdraw(void) : GlStatus
+ SetDescription(TextBlock) : void
+ GetDescription(void) : TextBlock
+ SetAdvGroups(StringVector) : void
+ GetAdvGroups(void) : StringVector
+ SetExpirationDate(DateTime) : void
+ GetExpirationDate(void) : DateTime
+ GetContact(void) : IoAdContact_C
+ SetContact(IoAdContact_C) : void
+ GetGuideURL(void) : String
+ SetGuideURL(String) : void
contact person
[PERSISTENT CLASS]
P
IoAdContact_C
+ myOrganizationName : String
+ myFaxNumber : String
+ myTelephoneNumber : String
+ myEMail : String
+ myZipCode : String
+ myCountryName : String
+ myStateName : String
+ myCityName : String
+ myStreetName : String
+ myLastName : String
+ myMiddleName : String
+ myFirstName : String
[PERSISTENT CLASS]
[DISTR OBJ]
P
IoAdServiceAdvertisement_C
_ myServiceUR : GlUR
_ myServiceClass : String
_ myServiceName : String
+ IoAdServiceAdvertisement_C(serviceName:String, serviceClass:String, serviceUR:glUR, advertiserUR,GlUR)
+ SetServiceUR(serviceUR:GlUR) : void
+ GetServiceUR(void) : serviceUR:GlUR
+ SetServiceClass(String) : void
+ GetServiceClass(void) : String
+ SetMime(mimeType:IoAdMimeType_C) : void
+ GetMime(void) : IoAdMimeType_C
SetSignature(signatureType:IoAdSignatureType_C)
+ GetSignature(void) : IoAdSignatureType_C*
+ SetServiceType(enum) : void
+ GetServiceType(String) : enum
+ SetServiceName(serviceName:String) : void
+ GetServiceName(String)
+ Update() : GlStatus
+ Submit() : GlStatus
isprovided by
[PERSISTENT CLASS]
[DISTR OBJ]
P
IoAdProviderAdvertisement_C
_ myProviderRole : String
_ myOrganizationName : String
_ myProviderAccessRestriction : String
+ IoAdProviderAdvertisement_C(advertiser"GlUR, role:String, orgName:String, accessRestrictions:String)
+ getProviderRole(void) : String
+ setProviderRole(role:String) : void
+ getOrganizationName(void) : String
+ setOrganizationName(orgName:String) : void
+ getAccessRestrictions(void) : String
+ setAccessRestrictions(accessRestrictions:String) : void
Submit()
Update()
provides
[PERSISTENT CLASS]
[DISTR OBJ]
P
IoAdProductAdvertisement_C
_ myProductTypeDescription : TextBlock
_ myProductTypeName : String
+ GetProductTypeDescription(void) : TextBlock
+ SetProductTypeDescription(prodDescription:TextBlock, : void productDescription:TextBlock, prodName:String)
+ IoAdProductAdvertisement(advertiser:GlUR)
Submit()
Update()
+ GetProductTypeName(void) : String
+ SetProductTypeName(prodName:String) : void
applies to
described by
ECSCollection
ECSSource
ECSInformationContent
has
[PERSISTENT CLASS]
P
IoAdServiceType_C
_ myServiceTypeName : String
+ IoAdServiceType_C(name:String)
+ GetServiceTypeName(void) : String
+ SetServiceTypeName(name:String) : void
[PERSISTENT CLASS]
P
IoAdMimeType_C
_ myURL : String
_ myMimeType : String
+ GetMimeType(void) : String
+ SetMimeType(String) : void
+ GetURL(void) : String
+ SetURL(String) : void
+ IoAdMimeType(url:String,mimeType:String)
[PERSISTENT CLASS]
P
IoAdSignatureType_C
_ mySignatureAttributes : GlParameterList
_ myRequiredSWList : IoAdRequiredSWVector
+ GetRequiredSWList(void) : IoAdRequiredSWVector
+ SetRequiredSWList(software:IoAdRequiredSWVector) : void
+ IoAdSignatureType(software:IoAdRequiredSWVectore, signatureAttr:GlParameterList)
+ GetSignatureAttributes(signatureAttr:GlParameterList) : void
+ SetSignatureAttributes(void) : GlParameterList
SignatureList
GlParameterList <RWVector>
has
[PERSISTENT CLASS]
P
IoAdRequiredSW_C
_ mySWName : String
_ myEarliestSWVersion : GlVersion
_ mySWUR : GlUR
_ myLatestSWVersion : GlVersion
+ GetSWName(swName:String) : void
+ SetSWName(void) : String
+ GetEarliestSWVersion(version:GlVersion) : void
+ SetEarliestSWVersion(void) : GlVersion
+ GetSWUR(swUR:GlUR) : void
+ SetSWUR(void) : GlUR
IoAdRequiredSW()
+ GetLatestSWVersion(version:GlVersion) : void
+ SetLatestSWVersion(void) : GlVersion

4.3.2-1. IoAd_SubmissionIF Object Model Diagram

Attributes:

None

Operations:

None

Associations:

The ECSCollection class has associations with the following classes:
Class: ECSInformationContent
Class: ECSSource
Class: IoAdProductAdvertisement_C describedby

4.3.2.2ECSInformationContent Class

Parent Class: Not Applicable
Public: NoDistributed Object: No
Purpose and Description:

Attributes:

None

Operations:

None

Associations:

The ECSInformationContent class has associations with the following classes:
Class: ECSCollection

4.3.2.3ECSSource Class

Parent Class: Not Applicable

Public: NoDistributed Object: No

Purpose and Description:

The ECSSource Class is a placeholder that represents the equivalent object from the ECS Core metadata model. All of the source specific attributes from the core will be contained in this object.

Attributes:

None

Operations:

None

Associations:

The ECSSource class has associations with the following classes:

Class: ECSCollection

4.3.2.4GlParameterList<RWVector> Class

Parent Class: Not Applicable

Public: NoDistributed Object: No

Purpose and Description:

The GIParameterList class is placeholder that represents the equivalent object from the global class category. This contains the advertisement submission list of parameters.

Attributes:

None

Operations:

None

Associations:

The GlParameterList<RWVector> class has associations with the following classes:
Class: IoAdSignatureType_C SignatureList

4.3.2.5IoAdAdvertisement_C Class

Parent Class: Not Applicable
Public: YesDistributed Object: Yes
Persistent Class: True
Purpose and Description:
This class provides the general attributes and operations for all advertisement interface objects. Specific derived types of advertisements (e.g., IoAdProductAdvertisement_C) inherit the properties of this base class.

Attributes:

myAdvGroups - stores the list of advertising groups to which this advertisement belongs.
Data Type: StringVector
Privilege: Private
Default Value:

myAdvertiserUR - stores the universal reference of the advertising DBMS application server.
Data Type: GlUR
Privilege: Private
Default Value:

myContact - stores the IoAdContact_C object that contains the human contact information for the advertisement.
Data Type: IoAdContact_C
Privilege: Private
Default Value:

myDescription - stores the text description of the advertisement.
Data Type: TextBlock
Privilege: Private
Default Value:

myExpirationDate - stores the expiration date for the advertisement.
Data Type: DateTime
Privilege: Private

Default Value:

myGuideURL - stores the URL of the hypertext guide document for the advertisement.
Data Type: String
Privilege: Private
Default Value:

Operations:

GetAdvGroups - this operation is used to get the list of advertising groups to which this advertisement belongs.
Arguments: void
Return Type: StringVector
Privilege: Public

GetContact - this operation is used to get the IoAdContact_C object that holds informatio about a human contact for the advertisement.
Arguments: void
Return Type: IoAdContact_C
Privilege: Public

GetDescription - this operation is used to get a copy of the current description of the advertisement.
Arguments: void
Return Type: TextBlock
Privilege: Public

GetExpirationDate - this operation is used to get the current expiration date of the advertisement.
Arguments: void
Return Type: DateTime
Privilege: Public

GetGuideURL - this operation returns the URL of the guide document for the advertisment.
Arguments: void
Return Type: String
Privilege: Public

GetIcon - this operation returns the bitmap of the icon used to represent the advertisement.
Arguments: void
Return Type: birmap
Privilege: Public

IoAdAdvertisement
Arguments: advertiser: GlUR

SetAdvGroups - this operation is used to set the list of advertising groups to which this advertisement belongs.
Arguments: StringVector
Return Type: void
Privilege: Public

SetContact - this operation is used to set a new IoAdContact_C object for the advertisement.
Arguments: IoAdContact_C
Return Type: void
Privilege: Public

SetDescription - this operation is used to set a new version of the description of the advertisement.
Arguments: TextBlock
Return Type: void
Privilege: Public

SetExpirationDate - this operation is used to set the expiration date for the advertisement.
Arguments: DateTime
Return Type: void
Privilege: Public

SetGuideURL - this operation is used to set a new URL for the guide document that describes the advertisement.
Arguments: String
Return Type: void
Privilege: Public

SetIcon - this operation is used to set a new bitmap to represent the advertisement.
Arguments: birmap
Return Type: Void
Privilege: Public

Submit - this operation is used to submit the advertisement to the advertising DBMS Application server.
Arguments: void
Return Type: GlStatus
Privilege: Public

Update - this operation is used to send an updated advertisement to the advertising DBMS

application server. Before the update operation is called, the advertising information should be modified via the appropriate setXXXX operations of this class.
Arguments: void
Return Type: GlStatus
Privilege: Public

Withdraw - this operation is used to withdraw an advertisement that was previously submited to the advertising DBMS application server.
Arguments: void
Return Type: GlStatus
Privilege: Public

Associations:

The IoAdAdvertisement_C class has associations with the following classes:
Class: IoAdContact_C contactperson

4.3.2.6IoAdContact_C Class

Parent Class: Not Applicable
Public: YesDistributed Object: No
Persistent Class: True
Purpose and Description:
The IoAdContact_C class manages access to the contact person for a given advertisement.

Attributes:

myCityName - stores the city name of the address of the contact person.
Data Type: String
Privilege: Public
Default Value:

myCountryName - stores the country name of the address of the contact person.
Data Type: String
Privilege: Public
Default Value:

myEMail - stores the e-mail address of the contact person.
Data Type: String
Privilege: Public

Default Value:

myFaxNumber - stores the fax phone number of the contact person.
Data Type: String
Privilege: Public
Default Value:

myFirstName - stores the first name of the contact person.
Data Type: String
Privilege: Public
Default Value:

myLastName - stores the last name of the contact person.
Data Type: String
Privilege: Public
Default Value:

myMiddleName - stores the middle name of the contact person.
Data Type: String
Privilege: Public
Default Value:

myOrganizationName - stores the organization name of the contact person.
Data Type: String
Privilege: Public
Default Value:

myStateName - stores the state name of the address of the contact person.
Data Type: String
Privilege: Public
Default Value:

myStreetName - stores the street name of address of the contact person.
Data Type: String
Privilege: Public
Default Value:

myTelephoneNumber - stores the telephone number of the contact person.
Data Type: String
Privilege: Public
Default Value:

myZipCode - stores the zip code of the address of the contact person.
Data Type: String
Privilege: Public

Default Value:

Operations:

None

Associations:

The IoAdContact_C class has associations with the following classes:
Class: IoAdAdvertisement_C contactperson

4.3.2.7IoAdMimeType_C Class

Parent Class: IoAdServiceType_C
Public: YesDistributed Object: No
Persistent Class: True
Purpose and Description:
The IoAdMimeType_C class provides access information for a service that is represented as a Mime type (e.g., http, telnet, etc.).

Attributes:

myMimeType - stores the name of the Mime type of the service.
Data Type: String
Privilege: Private
Default Value:

myURL - stores the URL of the service to be accessed.
Data Type: String
Privilege: Private
Default Value:

Operations:

GetMimeType - returns the name of the Mime type of the service.
Arguments: void
Return Type: String
Privilege: Public

GetURL - this operation returns the URL of the service.
Arguments: void
Return Type: String
Privilege: Public

IoAdMimeType - this constructor is used to initialize the IoAdMimeType_C object.
Arguments: url:String,mimeType:String
Return Type: Void
Privilege: Public

SetMimeType - this operation is used to set a new Mime type name for the service.
Arguments: String
Return Type: void
Privilege: Public

SetURL - this operation is used to set a new URL for the service.
Arguments: String
Return Type: void
Privilege: Public

Associations:

The IoAdMimeType_C class has associations with the following classes:
None

4.3.2.8IoAdProductAdvertisement_C Class

Parent Class: IoAdAdvertisement_C
Public: YesDistributed Object: Yes
Persistent Class: True
Purpose and Description:
The IoAdProductAdvertisement_C class is a specialization of the IoAdAdvertisement_C base class that is used to provide access to a product advertisement.

Attributes:

myProductTypeDescription - stores a text description of the product type.
Data Type: TextBlock
Privilege: Private

Default Value:

myProductTypeName - stores the product name.
Data Type: String
Privilege: Private
Default Value:

Operations:

GetProductTypeDescription - this operation returns a text description of the product type.
Arguments: void
Return Type: TextBlock
Privilege: Public

GetProductTypeName - this operation returns the product type name.
Arguments: void
Return Type: String
Privilege: Public

IoAdProductAdvertisement - this constructor is used to initialize the IoAdProductAdvertisement_C object.
Arguments: advertiser:GlUR
Return Type: Void
Privilege: Public

SetProductTypeDescription - this operation is used to set a new text description of the product type.
Arguments: prodDescription:TextBlock, productDescription:TextBlock, prodName:String
Return Type: void
Privilege: Public

SetProductTypeName - this operation is used to set a new product type name.
Arguments: prodName:String
Return Type: void
Privilege: Public

Submit - this operation is a specialization of the virtual operation in the IoAdAdvertisement_C base class. It is used to submit a new version of a product advertisement to the advertising DBMS application server.
Arguments:

Update - this operation is a specialization of the virtual operation in the IoAdAdvertisement_C base class. It is used toupdate a previously subbmited version of a

product advertisement to the advertising DBMS application server.
Arguments:

Associations:

The IoAdProductAdvertisement_C class has associations with the following classes:
Class: IoAdServiceAdvertisement_C appliesto
Class: ECSCollection describedby
Class: IoAdProviderAdvertisement_C provides

4.3.2.9IoAdProviderAdvertisement_C Class

Parent Class: IoAdAdvertisement_C
Public: YesDistributed Object: Yes
Persistent Class: True
Purpose and Description:
The IoAdProviderAdvertisement_C class is a specialization of the IoAdAdvertisement_C class that provides a representation of a provider advertisement.

Attributes:

myOrganizationName - stores the organization name of the provider.
Data Type: String
Privilege: Private
Default Value:

myProviderAccessRestriction - stores a text description of any access restrictions imposed by the provider.
Data Type: String
Privilege: Private
Default Value:

myProviderRole - stores the name of the role performed by the provider of the advertisement.
Data Type: String
Privilege: Private
Default Value:

Operations:

IoAdProviderAdvertisement_C

Arguments: advertiser"GlUR, role:String, orgName:String, accessRestrictions:String
Return Type: Void
Privilege: Public

Submit - this operation is a specialization of the virtual operation in the IoAdAdvertisement_C base class. It is used to submit a new provider advertisement to the advertising DBMS application server.
Arguments:

Update - this operation is a specialization of the virtual operation in the IoAdAdvertisement_C base class. It is used to update a previously submitted provider advertisement to the advertising DBMS application server.
Arguments:

getAccessRestrictions - this operation returns a text description of access restrictions imposed by the provider.
Arguments: void
Return Type: String
Privilege: Public

getOrganizationName - this operation returns the organization name of the provider.
Arguments: void
Return Type: String
Privilege: Public

getProviderRole - this operation returns the name of the role of the provider.
Arguments: void
Return Type: String
Privilege: Public

setAccessRestrictions - this operation set a new text description of access restrictions imposed by the provider.
Arguments: accessRestrictions:String
Return Type: void
Privilege: Public

setOrganizationName - this operation sets the organization name of the provider.
Arguments: orgName:String
Return Type: void
Privilege: Public

setProviderRole - this operation sets the role name of the provider.
Arguments: role:String
Return Type: void
Privilege: Public

Associations:

The IoAdProviderAdvertisement_C class has associations with the following classes:
Class: IoAdServiceAdvertisement_C isprovidedby
Class: IoAdProductAdvertisement_C provides

4.3.2.10IoAdRequiredSW_C Class

Parent Class: Not Applicable
Public: YesDistributed Object: No
Persistent Class: True
Purpose and Description:
This IoAdRequiredSW_C class provides access to information describing a software component required to access the service.

Attributes:

myEarliestSWVersion - stores the earlist useable version number of the software component.
Data Type: GlVersion
Privilege: Private
Default Value:

myLatestSWVersion - stores the latest useable version of the software component that is required to access the service.
Data Type: GlVersion
Privilege: Private
Default Value:

mySWName - stores the name of the software component.
Data Type: String
Privilege: Private
Default Value:

mySWUR - stores the universal reference of the software component storage location.
Data Type: GlUR

Privilege: Private
Default Value:

Operations:

GetEarliestSWVersion - this operation returns the earliest useable software version object.
Arguments: version:GlVersion
Return Type: void
Privilege: Public

GetLatestSWVersion - this operation returns the latest useable software version of the software component.
Arguments: version:GlVersion
Return Type: void
Privilege: Public

GetSWName - this operation returns the name of the software component.
Arguments: swName:String
Return Type: void
Privilege: Public

GetSWUR - this operation returns the universal reference of the location of the software component.
Arguments: swUR:GlUR
Return Type: void
Privilege: Public

IoAdRequiredSW - this constructor is used to initialize the IoAdRequiredSW_C object.
Arguments:

SetEarliestSWVersion - this operation is used to set a new earliest software version object.
Arguments: void
Return Type: GlVersion
Privilege: Public

SetLatestSWVersion - this operation is used to set a new version of the latest useable software version object.
Arguments: void
Return Type: GlVersion
Privilege: Public

SetSWName - this operation is used to set the name of the software component.

Arguments: void
Return Type: String
Privilege: Public

SetSWUR - this operation is used to set a new universal reference for the location of the software component.
Arguments: void
Return Type: GlUR
Privilege: Public

Associations:

The IoAdRequiredSW_C class has associations with the following classes:
Class: IoAdSignatureType_C has

4.3.2.11IoAdServiceAdvertisement_C Class

Parent Class: IoAdAdvertisement_C
Public: YesDistributed Object: Yes
Persistent Class: True
Purpose and Description:
This operation is a specialization of the IoAdAdvertisement_C base class. It is responsible for representing the interface to a service advertisement.

Attributes:

myServiceClass - stores the name of the service class for the advertised service.
Data Type: String
Privilege: Private
Default Value:

myServiceName - stores the name of the advertised service.
Data Type: String
Privilege: Private
Default Value:

myServiceUR - stores the universal reference for the service.
Data Type: GlUR
Privilege: Private
Default Value:

Operations:

GetMime - this operation returns the current IoAdMimeType_C object.
Arguments: void
Return Type: IoAdMimeType_C
Privilege: Public

GetServiceClass - this operation returns the current service class name of the advertised service.
Arguments: void
Return Type: String
Privilege: Public

GetServiceName - this operation returns the name of the advertised service.
Arguments: String
Return Type: Void
Privilege: Public

GetServiceType - this operation returns the current service type of the advertised service. Valid types are Mime and Signature.
Arguments: String
Return Type: enum
Privilege: Public

GetServiceUR - this operation returns the current universal reference of the advertised service.
Arguments: void
Return Type: serviceUR:GlUR
Privilege: Public

GetSignature - this operation returns the current service signature information via an IoAdSignatureType_C object.
Arguments: void
Return Type: IoAdSignatureType_C*
Privilege: Public

IoAdServiceAdvertisement_C - this constructor is responsible for initializing the IoAdServiceAdvertisement_C class.
Arguments: serviceName:String, serviceClass:String, serviceUR:glUR, advertiserUR,GlUR
Return Type: Void
Privilege: Public

SetMime - this operation is used to set a new IoAdMimeType_C object which holds information required to install a service that is accessed as a Mime type.
Arguments: mimeType:IoAdMimeType_C
Return Type: void
Privilege: Public

SetServiceClass - this operation is used to set a new service class name for the advertised service.
Arguments: String
Return Type: void
Privilege: Public

SetServiceName - this operation is used to set the name of the advertised service.
Arguments: serviceName:String
Return Type: void
Privilege: Public

SetServiceType - this operation is used to set the service type of the advertised service. Valid types are Mime or Signature.
Arguments: enum
Return Type: void
Privilege: Public

SetServiceUR - this operation is used to set a new universal reference for the service.
Arguments: serviceUR:GlUR
Return Type: void
Privilege: Public

SetSignature - this operation is used to set a new service signature via an IoAdSignatureType_C object.
Arguments: signatureType:IoAdSignatureType_C

Submit - this operation submits a new version of the advertisement to the advertising DBMS application server.
Arguments:
Return Type: GlStatus
Privilege: Public

Update - this operation is a specialization of the Update operation of the IoAdAdvertisement_C base class. This version of update handles the update of a IoAdServiceAdvertisement_C.
Arguments:
Return Type: GlStatus
Privilege: Public

Associations:

The IoAdServiceAdvertisement_C class has associations with the following classes:
Class: IoAdProductAdvertisement_C appliesto
Class: IoAdServiceType_C has
Class: IoAdProviderAdvertisement_C isprovidedby

4.3.2.12IoAdServiceType_C Class

Parent Class: Not Applicable
Public: YesDistributed Object: No
Persistent Class: True
Purpose and Description:
The IoAdServiceType_C class is a base class for the service type portion of a service advertisement.

Attributes:

myServiceTypeName - stores the name of the service type.
Data Type: String
Privilege: Private
Default Value:

Operations:

GetServiceTypeName - this operation returns the name of the service type.
Arguments: void
Return Type: String
Privilege: Public

IoAdServiceType_C - this constructor initializes the IoAdServiceType_C object.
Arguments: name:String
Return Type: Void
Privilege: Public

SetServiceTypeName - this operation is used to set a new service type name.
Arguments: name:String
Return Type: void

Privilege: Public

Associations:

The IoAdServiceType_C class has associations with the following classes:
Class: IoAdServiceAdvertisement_C has

4.3.2.13IoAdSignatureType_C Class

Parent Class: IoAdServiceType_C
Public: YesDistributed Object: No
Persistent Class: True
Purpose and Description:
The IoAdSignatureType_C class provides access to information required to access a signature type service. Signature type services are accessed via dynamic API services.

Attributes:

myRequiredSWList - stores the list of required software for the service.
Data Type: IoAdRequiredSWVector
Privilege: Private
Default Value:

mySignatureAttributes - stores the list of signature attributes for the service.
Data Type: GlParameterList
Privilege: Private
Default Value:

Operations:

GetRequiredSWList - this operation returns the list of required software for the service.
Arguments: void
Return Type: IoAdRequiredSWVector
Privilege: Public

GetSignatureAttributes - this operation returns the list of signature attributes for the service.
Arguments: signatureAttr:GlParameterList
Return Type: void

Privilege: Public

IoAdSignatureType - this constructor is used to initialize the mySignatureAttributes and myRequiredSWList attributes.
Arguments: software:IoAdRequiredSWVectore, signatureAttr:GlParameterList
Return Type: Void
Privilege: Public

SetRequiredSWList - this operation is used to set a new list of required software for the service.
Arguments: software:IoAdRequiredSWVector
Return Type: void
Privilege: Public

SetSignatureAttributes - this operation is used to set a new list of signature attributes for the service.
Arguments: void
Return Type: GlParameterList
Privilege: Public

Associations:

The IoAdSignatureType_C class has associations with the following classes:
Class: GlParameterList<RWVector> SignatureList
Class: IoAdRequiredSW_C has

4.4 CSCI Structure

The following table provides a summary of the components which make up this CSCI, to the extent they are currently known. Since this CSCI is on an incremental development, the table presents a current estimate of the CSCI components, but is possible to change as the CSCI evolves.

Table 4.4-1. ADSRV Components (1 of 2)

Name	Description	Type DEV - Developmental OTS - Of-the-shelf
AdvDBMSApplServer	Application server that processes the Advertising Client's requests (e.g., attribute search, create advertisement, and etc.).	DEV
AdvDBMSServer	DBMS used to store advertisement information.	OTS
AdvTextServer	Provides text search capability for Advertising Service.	OTS

Table 4.4-1. ADSRV Components (2 of 2)

Name	Description	Type DEV - Developmental OTS - Of-the-shelf
AdvNavigatingServer	Off the shelf http server to provides hyperlink access to advertisements.	OTS
GCMDExporter	Extracts information from Global Change Master Directory (GCMD) and generates ECS advertisements.	DEV

The following subsection provides a functional descriptions of the ADSRV components.

4.4.1. Advertising DBMS Application Server CSC (AdvDBMSApplServer)

4.4.1.1 Purpose and Description

The Advertising DBMS Application Server is a developed software component that processes the request from the interface objects described in Section 4.3. The interface objects communicate withe the AdvDBMSApplServer using DCE protocols. The server also uses the AdvDBMSServer for searching and persistant storage of the advertisements. This CSC will be developed by CIDM Release A Advertising Service personnel. .

4.4.2 Advertising DBMS Server CSC (AdvDBMSServer)

4.4.2.1 Purpose and Description

The Advertising DBMS Server CSC is a physical database server which stores and manages advertisement information. It provides the attribute search capability for users to locate interested advertisements.

4.4.2.2 Candidate Products

Sybase is the commercial-off-the-shelf (COTS) product selected for this CSC due to the considerations of L4 requirements and the DBMS Evaluations study. This decision is also synchronized with other ECS subsystems such as Data Management subsystem, CSMS/MSS subsystem, and etc.

A description of the database evaluation process and results can be found in the DBMS Evaluations Technical Paper (440-TP-002-001).

4.4.3 Advertising Text Server CSC (AdvTextServer)

4.4.3.1 Purpose and Description

The Advertising Text Server CSC provides the free text search capability for users to locate the interested advertisements in the ECS Advertising Service. Two approaches are currently under consideration:. The first approach is to use a commercial WAIS server (e.g. freeWAIS (Wide Area Information Server), CNIDR, WN Server). The second approach is to use theDBMS Server for performing text searches. Final resolution of the approach for this component will be determined via performance prototyping and evaluation of the capabilities which solution offers to the users.

4.4.4 Advertising Navigating Server CSC (AdvNavigatingServer)

4.4.4.1 Purpose and Description

The main purpose of Advertising Navigating Server is to allow Internet users to access ECS Advertising Service and be able to navigate through the advertisements. This CSC is a commercial-off-the-shelf (COTS) product which provides HTML/HTTP access to the Advertising DBMS Application Server.

4.4.5 Advertising GCMDExporter CSC (GCMDExporter)

4.4.5.1 Purpose and Description

The Advertising GCMDExporter imports the directory level information from the Global Change Master Directory via GCMD export files. The GCMDExporter does two levels mapping: 1) Schema mapping: maps GCMD DIF to Advertising Service Directory Attributes; 2) Valid mapping: reuses the ECS Gateway Mapping Service to do the valid mapping. For each GCMD DIF entry, the exporter will generate and submit one Data Product advertisement.

4.4.5.2 ECS white paper references

Full details of the exporter developement can be found on the Implementation Plan for the Release A Client Technical Paper (441-TP-001-001).

This page intentionally left blank.

Abbreviations and Acronyms

ACMHW	Access Control and Management HWCI
ADC	Affiliated Data Center
ADS	Archive data sets
ADSHW	Advertising Service HWCI
ADSRV	Advertising Service CSCI
AITHW	Algorithm Integration & Test HWCI
AITTL	Algorithm Integration and Test Tools (CSCI)
AM	Ante meridian
ANSI	American National Standards Institute
APC	Access/Process Coordinators
API	Application Programming Interface
APID	Application Process Identifier
AQAHW	Algorithm QA HWCI
ASAP	As soon as possible
ASCII	American Standard Code for Information Interchange
ASF	Alaska SAR Facility (DAAC)
ATM	Asynchronous Transfer Mode
CD ROM	Compact disk read only memory
CDRL	Contract Data Requirements List
CERES	Clouds and Earth's Radiant Energy System
CI	Configuration Item
CIESIN	Consortium for International Earth Science Information Network
CLS	Client Subsystem
COTS	Commercial off-the-shelf
CPU	Central processing unit
CSC	Computer Software Component
CSCI	Computer Software Configuration Item
CCSDS	Consultative Committee for Space Data Systems
CM	Configuration Management
CSDT	Computer Science Data Types
CSMS	Communications and Systems Management Segment
CSS	Communication Subsystem (CSMS)

DAA	DAN Acknowledge
DAAC	Distributed Active Archive Center
DADS	Data Archive and Distribution System
DAN	Data Availability Notice
DAO	Data Assimilation Office
DAR	Data Acquisition Request
DAS	Data Availability Schedule
DBA	Database administrator
DBMS	Database Management System
DDA	Data Delivery Acknowledgement
DDICT	Data Dictionary CSCI
DDIST	Data Distribution CSCI
DDN	Data Delivery Notice
DDSRV	Document Data Server CSCI
DESKT	Desktop CI
DEV	Developed code
DID	Data Item Description
DIM	Distributed Information Manager
DIMGR	Distributed Information Management CSCI
DIPHW	Distribution & Ingest Peripheral Management HWCI
DMGHW	Data Management HWCI
DMS	Data Management System
DMS	Data Management Subsystem
DP	Data Processing
DPR	December Progress Review
DPREP	Science Data Pre-Processing CSCI
DPS	Data Processing Subsystem
DR	Data Repository
DRPHW	Data Repository HWCI
DS	Data Server
DSM	Distribution Storage Management
DSS	Data Server Subsystem
DT	Data Type
ECS	EOSDIS Core System
EDC	EROS Data Center (DAAC)
EDOS	EOS Data and Operations System

EOC	Earth Observation Center (Japan)
EOS	Earth Observing System
EOSDIS	Earth Observing System Data and Information System
EP	Evaluation Package
EP	Early Prototype
ESDIS	Earth Science Data and Information System
ESDT	Earth Science Data Types
F&PRS	Functional and Performance Requirements Specification
FC	Fiber Channel
FDDI	Fiber distributed data interface
FDF	Flight Dynamics Facility
FOS	Flight Operations Segment
FSMS	File and Storage Management System
Ftp	File transfer protocol
GB	Gigabyte
GDAO	GSFC Data Assimilation Office
GFLOPS	Giga (billions) Floating Point Operations per Second
GOES	Geostationary Operational Environmental Satellite
GRIB	Gridded Binary
GSFC	Goddard Space Flight Center
GTWAY	Version 0 Interoperability Gateway CSCI
GUI	Graphic user interface
HDF	Hierarchical Data Format
HiPPI	High Performance Parallel Interface
HMI	Human machine interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transport Protocol
HWCI	Hardware Configuration Item
I&T	Integration and Test
I/O	Input/Output
ICD	Interface Control Document
ICLHW	Ingest Client HWCI
IDL	Interface Definition Language
IEEE	Institute of Electrical and Electronics Engineers
IERS	International Earth Rotation Service
IMS	Information Management Subsystem

IP	International Partner
IR-1	Interim Release 1
IRD	Interface Requirements Document
IS	Ingest Subsystem
ISS	Internetworking Subsystem (CSMS)
JPL	Jet Propulsion Laboratories
LaRC	Langley Research Center
LIM	Local Information Manager
LIMGR	Local Information Management CSCI
LIS	Lightning Imaging Sensor
L0	Level 0
MB	Megabyte
Mbps	Megabits per second
MBps	Megabytes per second
MD	Maryland
MFLOP	Millions of Floating Point Operations per Second
MOC	Mission Operations Center
MODIS	Moderate-Resolution Imaging Spectrometer
MPP	Massively Parallel Processor
MRF	Medium Range Forecast
MSFC	Marshall Space Flight Center
MSS	Management Subsystem (CSMS)
MTBF	Mean time between failures
MTTR	Mean time to restore
NESDIS	National Environmental Satellite Data and Information Service
NMC	National Meteorological Center
NOAA	National Oceanic and Atmospheric Administration
NSIDC	National Snow and Ice Data Center (DAAC)
O/A	Orbit/Attitude
ODC	Other Data Center
ODL	Object Description Language
ORNL	Oak Ridge National Laboratory (DAAC)
OSM	Open Storage Manager
OTS	Off-the-shelf
PAM	Permanent Archive Manager
PCI	Periphewral Component Interface

PDPS	Planning and Data Processing System
PDR	Preliminary Design Review
PDS	Production Data Set
PDS	Production Data Specialist
PGE	Product Generation Executive
PGS	Product Generation System
PLNHW	Planning HWCI
POSIX	Portable Operating System for UNIX
PRONG	Processing CSCI
Q	Quarter
Q/A	Quality Assurance
QA	Quality Assurance
QAC	Quality and Accounting Capsule
RAID	Redundant Array of Inexpensive Disks
RAM	Random Access Memory
REL	Release
RID	Review Item Discrepancy
RMA	Reliability, Maintainability, Availability
RTF	Rich Text Format
S/C	Spacecraft
SAA	Satellite Active Archives (NOAA)
SCF	Science Computing Facility
SCSI II	Small Computer System Interface
SDF	Software Development File
SDP	Science Data Processing
SDPF	Sensor Data Processing Facility (GSFC)
SDPS	Science Data Processing Segment
SDPS/W	Science Data Processing Software
SDPTK	SDP Toolkit CSCI
SDSRV	Science Data Server CSCI
SFDU	Standard Format Data Unit
SMC	System Management Center
SMP	Symmetric Multi-Processor
SPRHW	Science Processing HWCI
STMGT	Storage Management CSCI
TBD	To be determined

TBR	To be resolved
TDRSS	Tracking and Data Relay Satellite System
TONS	TDRSS Onboard Navigation System
TRMM	Tropical Rainfall Measuring Mission
TSDIS	TRMM Science Data and Information System
UR	Universal Reference
USNO	United States Naval Observatory
V0	Version 0
VC	Virtual Channel
VCDU-ID	Virtual Channel ID
WAIS	Wide Area Information Servers
WAN	Wide Area Network
WKBCH	Workbench CI
WKSHC	Working Storage HWCI
W/S	Workstation
WORM	Write Once Read Many
WS	Working Storage
WWW	World Wide Web